

PENERAPAN METODE DEPTH FIRST SEARCH PADA PENCARIAN RUTE BUS KOTA BERBASIS WEB MOBILE DI SOLO

Arif Herwanto¹, Bambang Eka Purnama²

¹STMIK AUB Surakarta, ²UNSA

ABSTRACT

Increasing the number of passengers and the needs of passengers over the years has resulted in many fleet of ground transportation, especially buses. Therefore the use of buses to transport more attractive, especially in solo. However, there are obstacles to a passenger if the passenger is to be given to the location of the destination and not knowing the destination so that passengers will find it difficult. And the use of mobile devices is very influential in the search for information for passengers because passengers can search for the information anywhere and anytime. In this thesis will be explained how to analyze and apply algorithms Depth First Search (DFS) in determining the city bus route and the shortest distance costs. DFS will look for all possible routes that may occur from the available data will then generate information on the cost and distance of the city bus service. The results of this study resulted in the city bus route search application based mobile web using PHP programming language.

Keywords: *DFS, bus routes, mobile web*

I. PENDAHULUAN

Di Kota Solo terdapat banyak jenis angkutan umum, salah satunya adalah bus kota, bus kota menjadi salah satu transportasi darat yang banyak digunakan masyarakat sebagai alternatif untuk berpergian, dikarenakan bus kota merupakan alat transportasi massal sangat mudah ditemukan. Namun dikarenakan banyaknya armada dari perusahaan jasa transportasi bus yang berbeda, biaya dan rute bus menjadi sangat beragam sehingga sangat menyulitkan bagi penumpang yang akan menggunakan jasa transportasi ini. Sebagai contoh, jika seorang masyarakat ingin berpergian ke suatu pusat perbelanjaan dan ternyata belum mengetahui jenis angkutan umum khususnya bus mana yang memiliki trayek melewati tujuan tersebut.

Dari salah satu sumber yaitu seorang pegawai yang bekerja dalam bidang perhubungan di kota Solo, beliau menyampaikan bahwa tidak sedikit calon penumpang yang merasa kebingungan mengenai bus kota yang akan digunakan untuk menuju tempat tujuan, dan dari permasalahan tersebut sangatlah dibutuhkan sebuah media yang bisa membantu calon penumpang dalam memudahkan mencari informasi mengenai rute dan tarif bus kota yang akan digunakan. Sebuah media yang berdaya guna harus berdasarkan pada sebuah algoritma pencarian yang benar juga agar permasalahan dapat diselesaikan secara tepat

sasaran. Algoritma pencarian Depth First Search yang sebagai blind search (pencarian buta) merupakan metode pencarian tanpa adanya informasi awal yang digunakan dalam proses pencarian. Hal ini sangat sesuai dengan pencarian solusi bagi penumpang dimana penumpang tidak mengetahui mengenai rute perjalanan dan tarif harga namun harus menentukan solusi dalam waktu yang terbatas. Namun, media tersebut tidak akan menjadi sebuah solusi jika informasi yang disediakan hanya bisa diakses melalui alat tertentu saja dan tidak bisa digunakan dimanapun berada. Maka dari itu, diperlukan sebuah sistem yang dapat membuat media tersebut menjadi fleksibel dan *mobile* (bergerak).

Berdasarkan penjelasan diatas, maka penulis berkeinginan untuk menggunakannya sebagai bahan penelitian dalam menyusun laporan skripsi dengan judul “Penerapan Metode Depth First Search Pada Pencarian Rute Bus Kota Berbasis Web Mobile Di Solo” dan akan di terapkan dalam Aplikasi Pencarian Rute Bus (APARUBUS).

II. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian dengan judul ” Metode *Depth First Search* Untuk Pencarian Rute Jadwal Penerbangan ”, oleh Maspin Sahputra, dari Universitas Sumatra Utara. Dalam penelitian tersebut membahas pencarian rute penerbangan untuk menampilkan semua solusi yang dapat terjadi sesuai dengan keinginan pemakai, baik dari segi waktu atau

biaya yang paling optimal dari kota 1 ke kota tujuan berdasarkan database dengan metode *Depth First Search*. Dalam penelitian tersebut, peneliti menggunakan teknologi pemrograman visual basic 6 dalam menerapkan metode pencariannya (Maspin Sahputra, 2012).

Yang kedua penelitian dengan judul “Pencarian Rute Terpendek dengan Menggunakan Algoritma *Depth First, Breath First* dan *Hill Climbing (Study Comparative)*” oleh Feddy Setio Pribadi dan Anggraini Mulwinda, Universitas Negeri Semarang, dalam penelitian tersebut peneliti melakukan penerapan 3 algoritma pencarian yaitu *Depth First, Breath First* dan *Hill Climbing* untuk penentuan rute terpendek yang paling optimum untuk dicapai. Peneliti menggunakan bahasa pemrograman bahasa C untuk menyelesaikan program *routing* (Feddy Setio Pribadi dan Anggraini Mulwinda, 2010).

Yang ketiga penelitian dengan judul “Implementasi Dan Analisis Algoritma *Depth First Search* (DFS) Dalam Pencarian Lintasan Terpanjang” oleh Sheila Eka Putri S, Universitas Sumatra Utara Medan, dalam penelitian ini peneliti menggunakan metode *depth first search* untuk memperoleh solusi optimum dari persoalan lintasan terpanjang pada *graph*. Hasil yang diperoleh merupakan suatu solusi optimum dengan nilai maksimum pada persoalan lintasan terpanjang pada *graph* yang juga dapat digunakan dalam menyelesaikan permasalahan *Hamiltonian Cycle* (sirkuit Hamilton) dan *Traveling Salesman Problem* (Persoalan Pedagang Keliling). Aplikasi dibangun berbasis desktop dengan pemrograman Java (penulis menggunakan NetBeans 6.9) (Sheila Eka Putri S, 2011).

Dari penelitian diatas, penelitian dengan judul “Metode *Depth First Search* Untuk Pencarian Rute Jadwal Penerbangan”, masih memiliki kekurangan yaitu waktu pencarian akan relatif lama jika data yang diproses terlalu banyak dikarenakan semua simpul yang memungkinkan akan disimpan kedalam memori, sehingga membutuhkan memori yang relatif besar. Dan pada penelitian dengan judul “Pencarian Rute Terpendek dengan Menggunakan Algoritma *Depth First, Breath First* dan *Hill Climbing (Study Comparative)*”, penelitian dilakukan

hanya difokuskan pada perbandingan algoritma dengan 3 algoritma yang digunakan untuk mencari algoritma terbaik dalam menyelesaikan sebuah permasalahan. Dalam penelitian ini, peneliti tidak melakukan implementasi, peneliti hanya melakukan pengujian terhadap sampel yang ada, tidak mempertimbangkan keefektifan dan keunggulan dalam penggunaan aplikasinya. Selanjutnya penelitian dengan judul “Implementasi Dan Analisis Algoritma *Depth First Search* (DFS) Dalam Pencarian Lintasan Terpanjang”, penelitian yang dilakukan menggunakan aplikasi berbasis desktop, dan perlu adanya pengembangan aplikasi untuk meningkatkan dan mengoptimalkan kegunaan aplikasi sebagai media pembelajaran.

Dari analisa penelitian diatas, penulis akan mencoba melakukan inovasi dan pengembangan dari beberapa kekurangan yang terdapat dalam penelitian terdahulu yang meliputi, aplikasi pencarian rute dibuat menggunakan teknologi web *mobile* dengan *jquery mobile* dimana teknologi yang akan penulis terapkan diharapkan dapat menjadi solusi dalam perkembangan jaman yang semua sudah berbasis *mobile* dan metode yang digunakan adalah metode *Depth First Search*, alat transportasi dikhususkan hanya dengan moda transportasi bus kota. Aplikasi ini dibuat untuk memudahkan para penumpang untuk memperoleh informasi mengenai rute dan biaya terbaik dalam melakukan perjalanan dalam kota dengan alat transportasi bus.

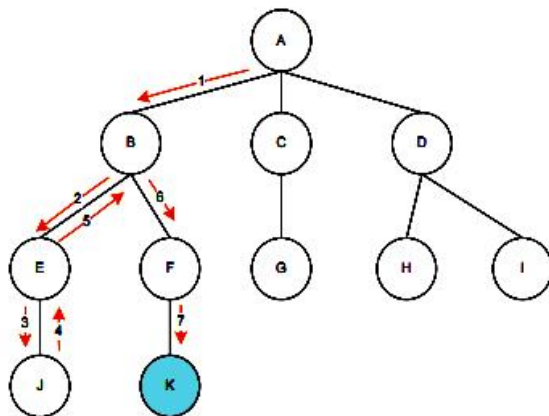
2.2. Sistem Transportasi

Transportasi adalah usaha memindahkan, menggerakkan, mengangkut, atau mengalihkan suatu obyek dari suatu tempat ke tempat lain agar lebih bermanfaat atau berguna untuk tujuan tertentu. Pengertian sistem transportasi secara operasional yaitu masalah masalah yang berhubungan dengan perpindahan barang dan atau orang (dengan memakai energi) yang bermanfaat bagi manusia (Kadir, 2006). Sedangkan pengertian transportasi kota adalah transportasi yang dilakukan di dalam suatu kota. Transportasi umum mempunyai permasalahan yang sama dalam hal ini mengenai kebingungan penumpang menentukan rutennya. Dalam sebuah kota besar dengan jaringan

transportasi yang rumit, penumpang sering tidak tahu bagaimana cara untuk mencapai tempat tujuan mereka kecuali tempat tersebut sering dikunjungi. Selain itu, setiap penumpang mungkin ingin merencanakan rute terdekat atau yang paling ekonomis untuk tujuan mereka.

2.3. Depth First Search (DFS)

Algoritma DFS pertama kali diperkenalkan oleh Tarjan dan Hopcroft 20 tahun lalu. Mereka menunjukkan bagaimana *Depth First Search* (DFS) merupakan metode pencarian secara mendalam dan bagian dari blind search atau pencarian buta. Pencarian dimulai dari level paling pertama, kemudian dilanjutkan ke anak paling kiri pada level berikutnya. Demikian seterusnya sampai tidak terdapat anak lagi atau level yang paling dalam. Jika pencarian belum menemukan solusi, maka dilakukan penelusuran kembali ke *node* sebelumnya dan dilanjutkan ke *node* tetangga. Proses ini diulangi terus hingga menemukan solusi (Kusumadewi, 2003).



Gambar 1. Pohon *Depth First Search*

- a. Keuntungan *Depth First Search*
 1. Membutuhkan memori yang relatif kecil, karena hanya *node – node* pada lintasan yang aktif saja yang disimpan.
 2. Secara kebetulan metode *Depth First Search* akan menemukan solusi tanpa harus menguji lebih banyak lagi dalam ruang keadaan.
- b. Kelemahan *Depth First Search*
 1. Memungkinkan tidak ditemukannya tujuan yang diharapkan.
 2. Hanya akan mendapatkan satu solusi pada setiap pencarian.

2.4. Konsep Mobile Web

Mobile web lebih mengarah pada akses ke *world wide web*, yaitu penggunaan browser berbasis layanan Internet, dari perangkat mobile, seperti *smartphone* atau *feature phone*, yang terhubung ke jaringan selular atau jaringan nirkabel lainnya. Aplikasi *web mobile* berjalan di dalam browser web pada perangkat mobile, aplikasi web mobile dibuat dengan HTML, CSS, dan JavaScript. Secara khusus, aplikasi web mobile berbeda dari situs web pada umumnya dalam hal tujuannya. Sebuah aplikasi web punya perhatian lebih pada user interface yang emniru aplikasi mobile natif. Ini tetap dibuat dengan teknologi web seperti HTML, CSS, JavaScript, dan AJAX tetapi menawarkan pengalaman yang mirip dengan sebuah aplikasi kepada penggunaanya (wahana-komputer, 2013).

III. METODE PENELITIAN

Menurut Sommerville, *waterfall model* meliputi kegiatan proses dasar spesifikasi, pengembangan, validasi, dan evolusi dan menyajikannya sebagai fase proses yang terpisah seperti spesifikasi persyaratan, desain perangkat lunak, implementasi, pengujian dan sebagainya. (Sommerville, 2011, h.30). Sommerville memecah model *waterfall* ini menjadi 5 tahapan meskipun secara garis besar sama dengan tahapan-tahapan model *waterfall* pada umumnya.

a. Analisis dan definisi kebutuhan (*Requirmnets Definition*)

Fitur, pembatas, dan tujuan dari sistem dibuat berdasarkan konsultasi dengan user. Hal – hal itu kemudian menjadi spesifikasi dari sistem tersebut.

b. Perancangan sistem dan piranti lunak (*System and Software Design*)

Proses ini mengalokasi kebutuhan – kebutuhan yang ada ke perangkat keras atau sistem perangkat lunak dengan membuat arsitektur sistem secara keseluruhan. Desain perangkat lunak melibatkan identifikasi dan penjelasan fundamental dari abstrak sistem dan hubungannya.

c. Implementasi dan *unit testing*

Pada tahap ini, desain perangkat lunak diwujudkan dalam bentuk unit – unit program. *Unit testing* bertujuan untuk memastikan bahwa setiap unit program dibuat sesuai dengan spesifikasinya.

d. Penggabungan dan pengujian sistem
(*Integration dan System Testing*)

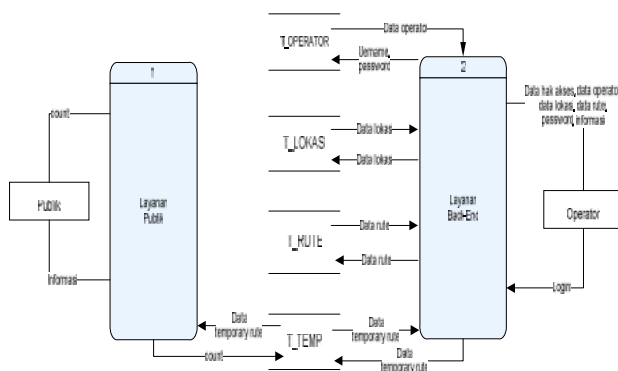
Unit – unit program di uji sebagai satu sistem yang utuh untuk memastikan bahwa semua spesifikasi telah dipenuhi. Setelah diuji, sistem dikirimkan ke *user*.

e. Pengoperasian dan pemeliharaan
(*Operation and Maintenance*)

Pada tahap ini tidak harus dilakukan, namun merupakan tahap yang kadang sangat diperlukan untuk melakukan pengembangan dan membutuhkan waktu paling lama. Pemeliharaan meliputi perbaikan kesalahan yang tidak ditemukan pada tahap – tahap sebelumnya, memperbaiki sistem dan menjadikan sistem lebih baik untuk memenuhi spesifikasi yang baru. Pemeliharaan perangkat lunak mengaplikasi lagi setiap fase sebelumnya, lalu memperbaiki program sebelumnya dan tidak membuat yang baru lagi.

3.1. DFD Level 1 APARUBUS

Pada DFD level 1 ini menunjukan lebih jauh keterlibatan entitas-entitas yaitu Operator, dan Publik. Pada gambar 2 tentang DFD level 1 tampak proses-proses seperti yang dijelaskan pada bagian fungsi-fungsi produk. Dimana fungsi tersebut adalah perwujudan dari proses yang melibatkan entitas untuk memasukkan dan menerima data dari sistem.



Gambar 2. DFD Level 1

Secara garis besar ada 2 proses yang akan secara langsung berinteraksi dengan entitas yaitu :

1. Proses Layanan Akses Publik

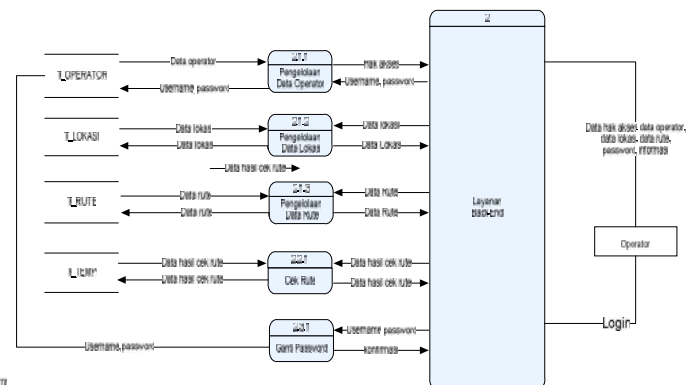
Proses ini menampilkan seluruh hasil pengelolaan aplikasi ke halaman publik sehingga dapat digunakan oleh pengguna.

2. Proses Layanan *Back-End*

Sedangkan proses *Back-End* memungkinkan seluruh operator berkolaborasi mengelola informasi sesuai dengan hak aksesnya sehingga dapat disajikan dengan cepat dan tepat kepada Member dan Publik.

3.2. DFD Level 2 Layanan *Back End*

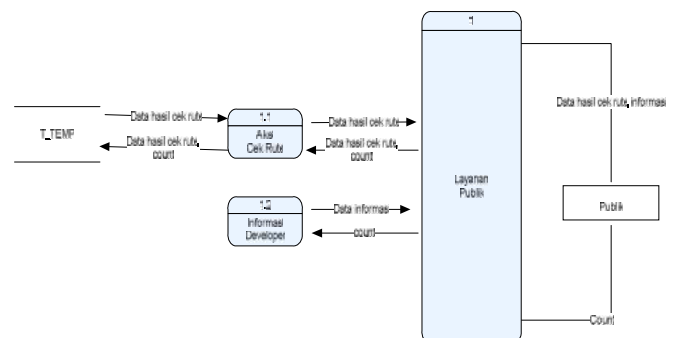
Proses ini menggambarkan detail struktur dari layanan back-end dimana Operator dapat mengelola data sesuai dengan hak akses. Layanan-layanan yang ada adalah layanan yang telah disebutkan pada bagian fungsi produk.



Gambar 3. DFD Level 2 Layanan *Back End*

3.3. DFD Level 2 APARUBUS

Proses ini menggambarkan detail struktur dari layanan publik dimana Publik dapat mengakses informasi yang ditampilkan APARUBUS sebagai berikut:



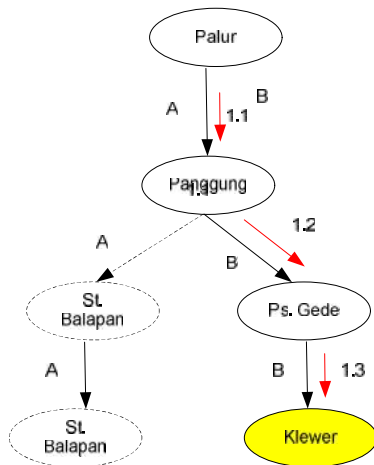
Gambar 4. DFD Level 2 Layanan Publik

3.4. Analisis Algoritma DFS

Sistem yang dibangun untuk menyelesaikan masalah, menggunakan metode pencarian *Depth First Search* (DFS). DFS akan mencari keseluruhan kemungkinan rute yang dapat terjadi dari data yang tersedia kemudian hasil dari seluruh pencarian akan dibandingkan untuk mendapatkan hasil yang tercepat atau termurah tergantung kondisi yang diinginkan. Pencarian solusi DFS dicirikan dengan simpul-simpul terdalam terdalam terlebih dahulu. Pertama simpul awal dibangkitkan, kemudian simpul pada arah kedua, pada simpul arah ketiga dan seterusnya. Jadi pencarian mengikuti sebuah lintasan tunggal mulai dari simpul awal terus menurun kebawah ke simpul-simpul arah bawahnya. Jika pencarian mencapai level terdalam, maka pencarian dilakukan level sebelumnya. Berikut adalah cara kerja algoritma DFS pada aplikasi pencarian rute bus kota. Contoh kasus yang dijelaskan dimisalkan yaitu rute bus dari palur ke klewer berdasarkan database.

A = Bus Atmo

B = Bus Damri



Gambar 5. Pencarian rute Palur – Klewer

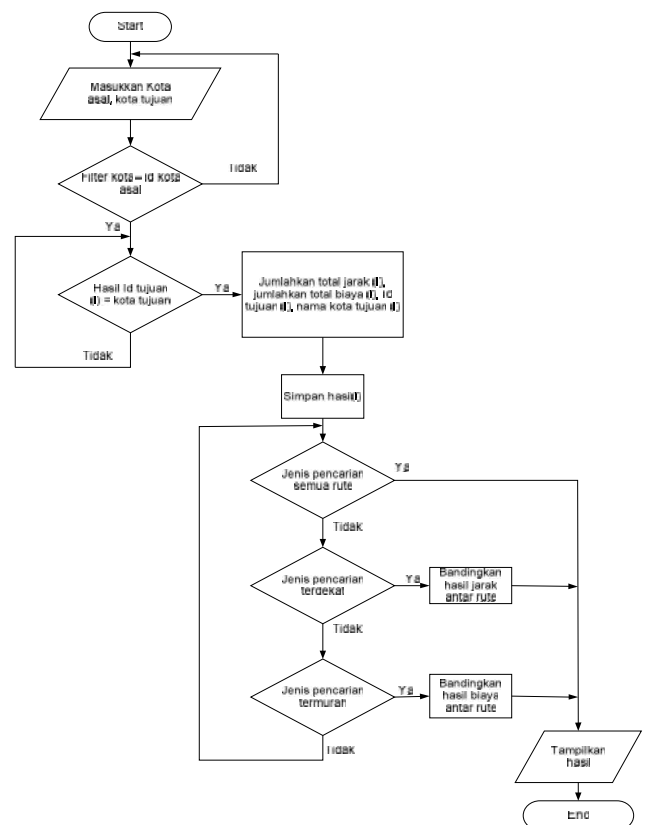
Pengecekan rute ke klewer , dimulai dari titik awal Palur menuju ke titik selanjutnya yaitu Panggung, pengguna dapat menggunakan bus Atmo dan Damri dan selanjutnya pada titik Panggung terdapat 2 cabang titik yaitu St. Balapan dan Ps.Gede, maka pengecekan dilakukan pada node paling kiri dulu, dikarenakan pada titik St. Balapan tidak ditemukan data Klewer maka kemudian pengecekan rute dilanjutkan pada node selanjutnya dan jika ditemukan data lokasi

sama dengan data lokasi tujuan maka pencarian dihentikan seperti yang ditunjukkan pada gambar 5, bus yang dapat digunakan yaitu bus Damri dari titik Panggung dan sehingga dapat dihasilkan data rute dari Palur menuju Klewer adalah seperti pada tabel hasil pencarian berikut :

Tabel 1 Hasil pencarian rute

No	Rute ke-1	Jarak	Biaya	Bus
1	Palur => Panggung	10.00	2500.00	Damri
2	Palur => Panggung	10.00	3000.00	Atmo
3	Panggung=>Pasar Gede	6.00	2500.00	Damri
4	Pasar Gede => Klewer	3.00	2000.00	Damri

3.5. Diagram Alir Algoritma



Gambar 6. Flowchart Algoritma

IV. HASIL DAN PEMBAHASAN

Implementasi digunakan untuk menjelaskan tentang manual modul kepada semua pengguna yang akan menggunakan sistem, sehingga pengguna dapat merespon apa yang ditampilkan dalam sistem dan memberikan masukan kepada pembangunan sistem untuk dilakukan perbaikan agar sistem

lebih baik lagi. Dan pada implementasi dikarenakan begitu banyaknya rute bus kota yang tersedia maka, data sampel yang digunakan adalah data rute dari Palur menuju Klewer meskipun aplikasi dapat diimplementasikan dengan rute apapun asalkan terdapat data jarak dan biaya. Seperti yang menjadi tujuan penelitian ini yaitu untuk merancang dan mengimple-mentasi aplikasi transportasi darat dengan metode *Depth First Search* (DFS) berbasis *web mobile* untuk rute perjalanan dengan bus kota untuk menentukan rute terdekat atau terpendek dan rute termurah.

4.1. Proses Pencarian Rute dengan DFS

Pada sub bab pencarian rute dengan metode DFS berisi salah satu contoh pencarian dengan sistem yang telah dibuat dengan data sampel seperti yang sudah dijelaskan pada sub bab sebelumnya yaitu rute Palur - Klewer. Contoh : seorang pengguna ingin melihat informasi rute bus kota dari Palur ke Klewer. Di tampilan sistem, user menginput data awal seperti pada gambar 7 kemudian dilakukan pencarian. Berikut langkah pencarian agar didapatkan *output* :



Gambar 7. Halaman Cari Rute

Setelah ditekan tombol cari oleh pengguna maka sistem akan melakukan proses filter dengan menampilkan hasil keseluruhan dari rute kota asal menuju kota tujuan seperti pada gambar 8, ditampilkan hasil pencarian rute disertai data jarak dan biaya per titik nya. Dan dilengkapi keterangan bus kota yang dapat digunakan bagi calon penumpang. Sehingga calon penumpang akan lebih mudah dalam menggunakan transportasi tersebut.

Cari rute bus dari kota : Palur

menuju ke : Klewer

✓ Cari

✗ Batal

Hasil Semua Pencarian adalah :

No.	Rute ke - 1	Jarak	Biaya	Bus
1	Palur ==> Panggung	10.00	2500.00	Damri
2	Palur ==> Panggung	10.00	3000.00	Atmo
3	Panggung ==> Pasar Gede	6.00	2500.00	Damri
4	Pasar Gede ==> Klewer	3.00	2000.00	Damri

Gambar 8. Tampilan hasil pencarian

4.2. Hasil Uji Algoritma DFS

Pada sub bab ini akan menyajikan perbandingan antara hasil perhitungan manual dengan hasil berdasarkan aplikasi yang digunakan untuk mengetahui apakah penerapan algoritma DFS sudah berjalan atau berfungsi seperti yang diharapkan dalam menentukan rute termurah dan terpendek.

Tabel 2. Hasil pencarian rute

No	Rute ke-1	Jarak	Biaya	Bus
1	Palur => Panggung	10.00	2500.00	Damri
2	Palur => Panggung	10.00	3000.00	Atmo
3	Panggung => Pasar Gede	6.00	2500.00	Damri
4	Pasar Gede => Klewer	3.00	2000.00	Damri

Dari hasil perhitungan secara manual yang dijelaskan dalam tabel 2 terdapat 3 rute dari Palur menuju Klewer yang bisa ditempuh, sesuai dengan titik awal calon penumpang berangkat dan berdasarkan aplikasi rute dari Palur menuju Klewer juga dapat ditempuh dengan 3 rute. Hasil dari Sistem yang dibuat juga menampilkan hasil yang sama seperti tampak pada gambar 9. Jadi dapat disimpulkan bahwa penerapan metode depth first search pada aplikasi pencarian rute bus (APARUBUS) handal atau sudah seperti yang diharapkan.

Hasil Semua Pencarian adalah :

No.	Rute ke - 1	Jarak	Biaya	Bus
1	Palur ==> Panggung	10.00	2500.00	Damri
2	Palur ==> Panggung	10.00	3000.00	Atmo
3	Panggung ==> Pasar Gede	6.00	2500.00	Damri
4	Pasar Gede ==> Klawer	3.00	2000.00	Damri

Gambar 9. Hasil pencarian dengan aplikasi

V. KESIMPULAN

Setelah melakukan pembahasan pada bab sebelumnya, maka penulis dapat memperoleh kesimpulan sebagai berikut :

- Depth First Search akan mencari semua solusi yang dapat terjadi pada semua rute dimulai dari simpul paling kiri dengan level paling tinggi sampai ke simpul paling kanan dengan level terdalam.
- Hasil pencarian dengan metode Depth First Search (DFS) dapat menampilkan semua solusi yang dapat terjadi sesuai dengan keinginan pemakai, baik dari segi jarak, atau biaya yang paling optimal.
- Pemanfaatan algoritma Depth First Search (DFS) untuk menemukan jalur-jalur alternatif ketika melakukan perjalanan untuk mengunjungi beberapa kota sehingga perjalanan yang dilakukan dapat optimal baik dari segi waktu maupun biaya.

DAFTAR PUSTAKA

- Abdul, Kadir. 2006. Transportasi: Peran dan Dampaknya dalam Pertumbuhan Ekonomi Nasional. Jurnal Perencanaan dan Pengembangan Wilayah Wahana Hijau. Vol. 1, No. 3.
- Abdul Kadir. 2008. Tuntunan Praktis Belajar Database Menggunakan MySQL, Yogyakarta : Andi.
- Akerkar, Rajendra. 2005. *Introducing to Artificial Intelligence*, Pretince-Hall, Hindia,
- Eka Putri Sheila. 2011. Implementasi Dan Analisis Algoritma *Depth First Search* Dalam Pencarian Lintasan Terpanjang. Universitas Sumatra Utara. Medan.
- Hakim, Lukamnul. 2010. Bikin Website Super Keren Dengan PHP dan JQuery. Yogyakarta : Lokomedia.
- http://en.wikipedia.org/wiki/Mobile_device, jam 2:19 tanggal 28 Agustus 2013.
- <http://id.wikipedia.org/wiki/Notepad%2B%2B>, jam 8:42 tanggal 12 juni2013.
- Jogiyanto, MBA, Ph.D. 2005. Analisis dan Desain Sistem Informasi, Yogyakarta : Andi.
- Kusumadewi, Sri. 2003. “ *Artificial Intelligence – Toeri dan Aplikasinya*”, Yogyakarta : Graha Ilmu.
- Raharjo Budi, dkk. Modul Pemrograman WEB (HTML, PHP, & MySQL), Modula. Bandung.
- Pressman, Roger, S. 2010. *Software Engineering : A Practitioner's Approach. 7th Edition*.
- Sahputra Maspin. 2012. Metode *Depth First Search* Untuk Rute Jadwal Penerbangan. Universitas Sumatra Utara. Medan.
- Setio Fredy dan Mulwinda Anggraini. 2010. Pencarian Rute Terpendek dengan Menggunakan Algoritma *Depth First Search, Breadth First Search* dan *Hill Climbing (Study Comparative)*. Universitas Negeri Semarang. Semarang.
- Sommerville, Ian. 2011. *Software Engineering*.
- Susanto, Azhar, MBus, Ak. 2004. *Sistem Informasi Manajemen, Konsep dan Pengembangannya*, Bandung : Lingga Jaya.
- Suyanto. 2007.“ *Artificial Intelligence*”, Bandung : Informatika.